Medicon Engineering Themes
Volume 5 Issue 2 August 2023
Article Type: Review Article
ISSN: 2834-7218

Comparative Analysis of Software Tools Used for BCI: The BCI Kit

Bushra Khatoon Zaidi[1]* and Atifa Arshad[2]

[1]*Mangalayatan University, Aligarh, India*

[2]*University of Delhi, India*

***Corresponding Author:** Bushra Khatoon Zaidi, Mangalayatan University, Aligarh, India.

Received: July 19, 2023; **Published:** July 27, 2023

DOI: 10.55162/MCET.05.155

Abstract

Brain computer interfaces (BCIs) are the technologies that transforms commands for operating devices from electrical activity in the brain. Most BCI systems consist mostly of hardware and software. Hardware is gathering the data while software interprets the brain's input and feeds it back to the hardware. In this article, we give a summary of the publicly accessible software platforms for brain-computer interfaces. Five significant BCI platforms have been determined by us. We outline the intended target user group (which consists of academics, programmers, and end users), as well as the key characteristics of each platform, including their compatibility with various operating systems, licensing, supported devices, and programming languages. BCI2000, OpenViBE, TOBI Common Implementation Platform (CIP), BCILAB, and BCI++ are the five platforms under question. In our conclusion, we address potential synergies and potential future advances, such as merging diverse platform components. With this analysis, we intend to highlight the benefits and drawbacks of each platform that is currently accessible, which should aid anyone working in the field of BCI research in choosing the best platform for their individual needs.

Keywords: bci; bci2000; bcilab; tobi; eeg

Introduction

Brain-Computer Interface (BCI) is a technology that enables a person to control an external device using brain signals. BCI exploration has concentrated on biomedical operations, similar as helping people impaired by a stroke, physical injury, or neurological complaint. BCIs could prop people with disabilities and ameliorate public defense capabilities, among other uses. There are colorful software tools and platforms available for experimenters to be used for BCI exploration, some of which includes BCI2000, OpenViBE, TOBI, BCILAB, BCI. A brief relative analysis of Software with a range of features, similar as source localization, shaft discovery, artefact rejection, diapason analysis, and connectivity assessment capabilities is also studied.

BCI Tools and their Applications

BCI2000

BCI2000 is a software suite for brain- computer interface (BCI) exploration. It's an open- source, general- purpose software system that's free for non-commercial use. BCI2000 is generally used for data accession, encouragement donation, and brain monitoring operations. It supports a variety of data accession systems, brain signals, and study/ feedback paradigms. During operation, BCI2000 stores data in a common format (BCI2000 native or GDF), along with all applicable event labels and information about system configuration. BCI2000, when was first used in July 2001 (5), it successfully was suitable to transmit brain signals over the internet. BCI2000 includes software tools that can acquire and reuse data. A general- purpose software platform for brain- computer interface (BCI) exploration is

called BCI2000. BCI2000 comes with software tools that can gather and reuse data, deliver feedback and stimulants, and control commerce with extraneous objects like robotic arms. The real- time BCI2000 system can synchronize EEG and other signals with a range of bio-signals and input tools, including mouse or eye- trackers. To manage data importing and exporting in popular train formats, it offers a number of modules. utmost Windows systems can run BCI2000, and utmost Windows computers can also be used to induce the source law.

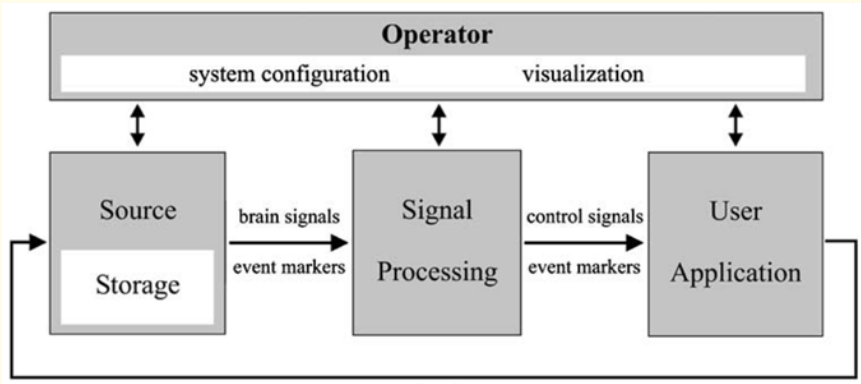

Figure 1: A schematic block diagram of the BCI2000 system. BCI2000 design. BCI2000 consists of four modules: operator, source, signal processing, and application. Operator module acts as a central relay for system configuration and online presentation of results to the investigator. It also defines onset and offset of operation. During operation, information (i.e., signals, parameters, or event markers) is communicated from source to signal processing, to user application, and back to source.

Open-ViBE

OpenViBE is a software platform dedicated to designing, testing and using brain-computer interfaces. OpenViBE is a form of software for real-time neurosciences, or for processing brain signals in real-time. It can be used to acquire, filter, process, classify and visualize brain signals in real time. OpenViBE now comes with a tool for offline or batch analysis of big datasets as of version 2.2.0 (6). It is a free and open source software. It works on Windows and Linux operating systems. OpenVIBE is distinguished by its attractive GUI interface, which makes creating a BCI application simple and quick. The platform has many modules for neuro-feedback and virtual reality. The framework can accommodate new modules as plug-ins. A data collection server is used to collect EEG data, which is then transferred over TCP/IP.

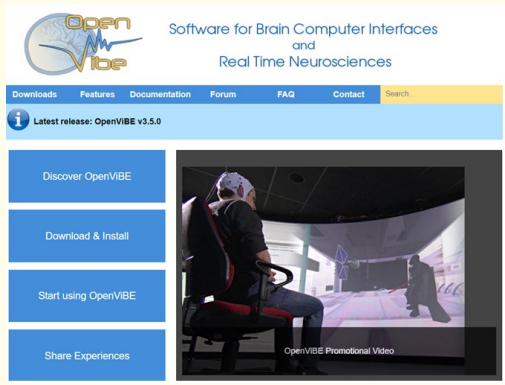

Figure 2: Open-ViBE Interface.

TOBI Common Implementation Platform (Python PyTIAClient)

The TOBI Common Implementation Platform (CIP) is a software platform for developing and evaluating brain-computer interface (BCI) systems. It provides a standardized interface for communication between different BCI components, such as data acquisition, feature extraction, classification, and feedback.

The Python PyTIAClient is a Python implementation of the TOBI Interface for Acquisition Client (TIAC) that allows for the acquisition of EEG data from different EEG amplifiers and devices. It provides a simple and easy-to-use interface for connecting to EEG devices and streaming data to other BCI components. Here's an example of how to use the Python PyTIAClient to connect to an EEG device and stream data:

```
python
from pytiac import PyTIAClient

# Create a PyTIAClient object
client = PyTIAClient()

# Connect to the EEG device
client.connect("localhost", 12345)

# Start streaming data
client.start()

# Read data from the EEG device
while True:
    data = client.read()
    print(data)

# Stop streaming data
client.stop()

# Disconnect from the EEG device
client.disconnect()
```

This code creates a PyTIAClient object and connects to an EEG device running on the local machine at port 12345. It then starts streaming data from the device and reads the data in a loop. Finally, it stops streaming data and disconnects from the device.

BCILAB

A MATLAB toolbox for BCI (brain-computer interface) analysis is called BCILAB. The toolbox's purpose is to give the BCI community a potent toolset for technique research and assessment, accelerating the rate of innovation in the industry. The toolbox facilitates the design and development of new methods for cognitive state classification, feature extraction, and feature selection. It may be used to examine openly accessible data from previous BCI contests and from a task requiring quick serial visual display. The BCILAB toolbox is complementary to the existing spectrum of tools for real-time BCI experimentation, deployment, and use (Kothe and Makeig, 5). It is used to classify EEG data using BCI both online and offline.

Wearable Sensing has created a DSI-Streamer to BCILAB plug-in that enables BCILAB to handle a real-time stream of data from DSI-Streamer through its TCP/IP socket in order to classify EEG using online BCI. BCILAB then handles the data in an efficient manner. DSI-Streamer files can be imported into EEGLAB using DSI-Streamer to EEGLAB extension (as seen below) and then exported as.set files for Offline BCI classification. After that, BCILAB may read the exported.set files. This matlab plugin for EEGLab provides a foundation for the creation of BCI applications and prototypes by enhancing the already available functionality.

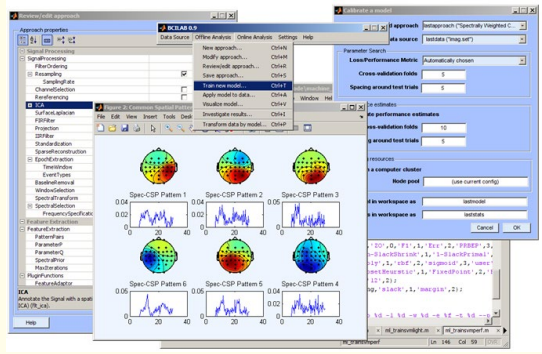

Figure 3: Sample subset of BCILAB's graphical user interface panels showing the main menu, a model visualization, a model configuration dialog (left), an evaluation setup dialog (right), and the script editor.

Platform	Windows	Mac OS X	Linux	License	Requirements
BCI2000	•	• 1	• 1	GPL	Windows [2]
OpenViBE	• 3	•	•	LGPL [4]	-
TOBI	•	• 5	•	GPL, LGPL 5	-
BCILAB	• 7	• 7	• 7	GPL	MATLAB [8]
BCI++	•	• 9	• 9	GPL	Windows [9]

Table 1: Feature comparison of BCI platforms.

Columns 2-4 indicate if operating systems are officially supported. Support for Windows includes versions XP, Vista, and 7 unless otherwise noted. Support for Mac OS X includes versions 10.5 and 10.6 unless otherwise noted. If Linux is supported, the platform should run on any Linux distribution. The last column lists all required software components that are not open source or freely available.

1. Officially supported in next version, current version should run under Mac OS X and Linux.

2. Next version will also run under Mac OS X and Linux.
3. Also runs on Windows 2000.
4. Version 2 or later.
5. Unofficially, the TOBI library runs on Mac OS X and iOS platforms.
6. TiA (TOBI Interfaces) is licensed under the LGPL; the TOBI Signal Server is licensed under the GPL.
7. All versions that run MATLAB R2006a or greater.
8. MATLAB is not required to run BCILAB, only for making changes at the source code level.
9. Unofficially, also runs and compiles on Mac OS X and Linux.

BCI++

BCI++ is a complex graphics engine-based open source framework. The platform offers a selection of instruments for the quick development of brain-computer interfaces and general human-computer interaction (HCI). Two primary modules make up the BCI++ framework, and they communicate with one another using TCP/IP. The first module, known as HIM (Hardware Interface Module), is in charge of real-time processing, signal storage, display, and signal acquisition. AEnima, the second module, offers a Graphical User Interface (GUI). Based on an advanced 2D/3D graphics engine, this module is responsible for developing and overseeing several protocols. With the use of this framework, a real-time BCI system's development was divided into two phases: (1) signal processing algorithms, and (2) a graphical user interface (GUI).

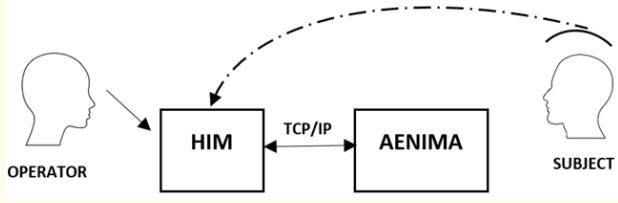

Figure 4: Structure of BCI++.

Future Directions

Exploiting synergies and reducing redundancies between platforms could be future directions. This ambition, as well as less ambitious short-term goals like making the platforms talk to one another, may be accomplished in large part because to the TOBI CIP. In doing so, it would be possible to combine the data collecting capabilities of one platform with the feature extraction capabilities of another platform and the visualization powers of a third framework. The modification of existing platforms to include the TOBI interfaces in addition to their native data interchange formats should be rather simple. The ability to adopt this standardized format offers up a wealth of potential, even if platform-specific capabilities had to be abandoned due to a lack of support in the TOBI protocols.

Summary and Conclusions

Over the past few years, there have been a lot more BCI software platforms available that are user-friendly. The day when a researcher had to design every necessary BCI component from scratch is almost finished, or at the very least, there are workable alternatives accessible. Nowadays, there are numerous publicly accessible BCI platforms available to those who want to utilize or create BCIs. In this article, five of the most well-liked BCI frameworks have been discussed. While certain platforms, like BCI2000 (5) and OpenViBE, have been around for a while and offer a tons of features, each platform has its own special features and advantages. The concerns like licensing, platform compatibility, that would be significant to potential consumers have been addressed. Table 1 compares all platforms in terms of supported operating systems, licenses, and specifications. It's interesting to note that every platform uses the GPL or LGPL as their license. Additionally, the majority of platforms, utilize multiple operating.

In conclusion, there is probably no one platform that works best for everyone. With the knowledge provided in this article, interested users should be able to recognize platforms that might be appropriate for their particular needs.

References

1. Kothe Christian Andreas and Scott Makeig. "BCILAB: a platform for brain-computer interface development". Journal of neural engineering 10.5 (2013): 056014.
2. Kögel J, Jox RJ and Friedrich O. "What is it like to use a BCI? - insights from an interview study with brain-computer interface users". BMC Med Ethics 21.2 (2020): 2.
3. U.S. Governmental Accountability Office. "Brain-Computer Interfaces". GAO-22-106118.
4. MyNewsdesk. "Brain-Computer Interface Market Share, Size Raising to USD 2,564 Million by 2024 with Noteworthy CAGR 15.1%". Ameco Research on Global Brain Computer Interfaces Market.
5. BCI2000. "System Features" National Centre for Adaptive Neurotechnologies".
6. Admin. "Discover Openvibe". OpenViBE (2018).
7. "The Core Components". Tools4bci (2023).

Volume 5 Issue 2 August 2023